AD-A078 413 MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR COMPUTE--ETC F/G 6/5
DESIGN OF A PROGRAM FOR EXPERT DIAGNOSIS OF ACID BASE AND ELECT--ETC(U)
MAY 79 R S PATIL N00014-75-C-0661
UNCLASSIFIED MIT/LCS/TM-132 NL

1 OF 1
AD-
A078413

END
DATE
FILMED
8-80
DTIC

LABORATORY FOR COMPUTER SCIENCE

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

MIT/LCS/TM-132

DESIGN OF A PROGRAM FOR EXPERT DIAGNOSIS OF ACID BASE AND ELECTROLYTE DISTURBANCES

Ramesh S. Patil

May 1979

545 TECHNOLOGY SQUARE, CAMBRIDGE, MASSACHUSETTS 02139

C

14 MIT/LCS/TM-132

6 Design of a Program for Expert Diagnosis of Acid Base and Electrolyte Disturbances,

10 Ramesh S. Patil

12 42

11 May 1979

This investigation was supported (in part) by National Institutes of Health Grant No. 1 P41 RR 01096-02 from the Division of Research Resources.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LABORATORY FOR COMPUTER SCIENCE

15 N00014-75-C-0661

CAMBRIDGE ~~MASSACHUSETTS 02139~~

Abstract

This research develops the diagnostic component of an interactive system for providing expert advice for the diagnosis, therapy and ongoing management of patients with acid-base and electrolyte disturbances. We have developed a hierarchic representation of a patient's illness which unifies the known facts about the patient, their suspected interrelationships, the hypotheses and how hypotheses account for various known and hypothesized facts. An expectation driven problem solver based on the hypothesize and reformulate paradigm performs the diagnosis.

KEY WORDS: Medical Diagnosis

Acknowledgements

I am grateful to Dr. William B. Schwartz and Prof. Peter Szolovits for introducing me to the area of clinical decision making and for their guidance, collaboration and encouragement in my research. I would also like to thank the members of Clinical Decision Making Group, especially Brian Smith, Bill Swartout, Byron Davies and Harold Goldberger for many extended discussions and criticisms.

Revised version of a Ph. D. thesis proposal submitted to the Department of Electrical Engineering and Computer Science

CONTENTS

1. Introduction

In recent years there has been a growing interest in the artificial intelligence community in the application of AI techniques to 'real-world' problems. The move towards what have been called knowledge-based systems represents a change from previous attempts at developing generalized problem solvers (as, for example, GPS [10]). Earlier work on these systems, while developing a large body of useful techniques (e.g., problem decomposition into subgoals, heuristic search etc. [10, 24]) have demonstrated a need for both the accumulation of large amounts of domain-specific knowledge and the development of domain-specific techniques in order to achive a high level of expertise.

In this paper we propose methods of representation and design for the application of knowledge based techniques to one important domain -- clinical medicine. Specifically, we propose to design a program for providing expert assistance for the diagnosis, therapy and ongoing management of patients with acid base and electrolyte disturbances.

The design of an expert medical system presents us with an excellent domain for the further development and testing of knowledge based AI techniques and the challenge of a socially useful, real world application. Attempts to formalize and implement theories of expert clinical skills focus on many AI issues: what is the nature of expert medical knowledge, how much knowledge is required and, how should it be organized and applied? We believe that medicine constitutes a promising application domain for the development and evaluation of AI techniques because of its highly developed taxonomy, generally repetitive problem-solving encounters, large number of case studies and acknowledged experts. Further, we believe that the application of computers to medicine will help alleviate some of the problems facing the health care system today: the quality of heath care, the high cost of medical training and the maldistribution of health services.

The ever-expanding body of knowledge in medicine has created a situation in which it is increasingly difficult for even the most conscientious physician to maintain a high level of competence across the range of medical problems he is likely to encounter. The resulting adverse effect on the quality of care is substantial. Therefore, any effort which could make expert consulting skills readily available to the practicing physicians should have a considerable impact on the quality of care delivered.

Medical education and training places a great deal of emphasis on experience as the principal means for developing the "feel" or expertise needed to become a capable doctor. With the availability of computer programs, a student could interact extensively with expert computer programs to learn clinical skills and test them on simulated patients.

Expert physicians simply do not wish to live and practice in the rural communities and the depressed inner cities, with the result that the health services suffer. Over the long run, it seems likely that adequate services in such areas can meet desired standards if the health care personnel practicing in these communities have ready access to expert consultation. Availability of expert advice through computers (including suggestions on when to refer a patient to larger facilities) should markedly improve the performance of both physician and non-physician medical personnel not well versed in the given area of expertise.

1.1 Domain of Expertise

We have chosen the domain of acid base and electrolyte disturbances for the following reasons. It is a relatively narrow area of medicine with a large body of physiological and clinical knowledge, with a high degree of interaction between them. It is an essential component in the diagnosis of the renal diseases, which has been the major area of medical expertise explored at our laboratory and is the area dealt with in the Present Illness Program [27, 28, 46]. Thus, the techniques developed in this area can be usefully extended to a general Present Illness Program. Further, the temporal characteristics of acid base and electrolyte disturbances provide us with a variety of issues involved in patient management. Lastly, programs in the same medical area but relying on older computer methodologies have existed for a number of years [2, 5, 14, 34] and can serve as both a realistic guide for minimum performance requirements for our eventual program and a point off comparison for investigating the differences of style dictated by our A/I approach.

At the outset, we would emphasize that our concern is not with acid-base and electrolyte disturbances per se. Our basic purpose is to use this domain to determine whether it is now possible to develop an "artificial intelligence" program which is capable of dealing with all aspects of diagnosis and management at a level comparable to that of an expert.

1.2 Scope of the Project

The objective of a program in the diagnostic and treatment domain is "the proper management of the patient." That proper management consists of collecting the relevant information about the patient, identifying the disease process(es) responsible for the patient's illness, and prescribing a proper course of actions to correct the patient's condition. One of the complexities of the task is that its subcomponents do not have well defined boundaries. This is because the patient may be presented to a clinician at different stages of a diseases evolution and treatment. During the course of management new information about the past history may become necessary as the diagnostic hypotheses evolve. The current diagnosis may also depend on information that is presently unavailable (e.g., serum-electrolytes that have been drawn but not reported by the laboratory). Moreover, the disease itself may evolve through time providing additional clues to its identity. Further, at times the response to a certain treatment itself may be the best clue to the diagnosis. Therefore, the clinician must choose the next course of action from a large range of alternatives, which can be broadly classified as gathering more information (much of which may turn out to be irrelevant in the current clinical evaluation), ordering further tests (involving possibly expensive time delays and/or clinical costs), waiting for further development, prescribing therapy or some combination of the above. Thus at every stage of expert consultation the program must be able to choose between the alternative actions with the objective of maximizing the utility to the patient. This can be achieved in a computer program only by developing a system capable of diagnosis, therapy and making decisions between various alternatives available to a physician during patient care. With this in view we propose to design the Acid/Base and Electrolyte Consultant system which will address each of the above mentioned aspects of diagnosis and management. Further, in keeping with our objective, we have tried to separate and modularize different components of a physician's knowledge and expertise so as to be able to evaluate our understanding about each and their interactions. This should also allow us to further experiment with, redesign, and implement any component of the system without having to reimplement the entire program again.

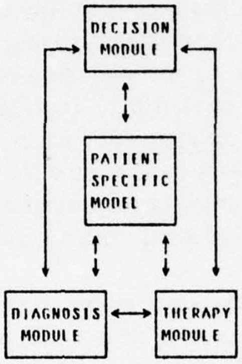


Fig. 1.1. A Schematic for the Overall ABEL System

The Acid/Base and Electrolyte Consultant system will therefore consist of four major components: i) the Patient Specific Model, ii) the Global Decision Making component, iii) the Diagnostic component, and iv) the Therapy component. The patient specific model describes the physician's understanding of the state of the patient at any point during the diagnosis and management. This patient specific model is used as a central data structure with which other components of the system may reason. The global decision making component is the top level program which has the responsibility of calling the other programs with specific tasks. In general the global decision program will call the diagnostic program with a task such as: take the initial history, elaborate some specific diagnosis, etc. The diagnosis component then performs the specified diagnostic task and reports the results to the main program. It also modifies the patient specific model to reflect the revised state of the patient. Similarly if the global decision making program calls the therapy selection program, it attempts to formulate a set of alternate therapies for the patient along with a check list of items that must be tested before any specific therapy can be recommended. It also identifies information that will help discriminate between alternate therapy recommendations. This information then is reformulated and sent to the diagnostic program as the next problem to be solved. This approach of separating diagnostic, decision-making and therapeutic activities allows us to make explicit the decision making that goes on in a physician's reasoning: e.g., is further diagnosis necessary, what treatment should be selected, should we wait before prescribing further treatment, can we choose some therapeutic action that would also provide diagnostic information thus making further diagnosis at this point unnecessary, etc.

Although the therapeutic capability of the program should ultimately be the most important from a clinical viewpoint, it is the development of the patient specific model and the diagnosis subprogram which first claims our attention. Because diagnosis must encompass at least past evidence of therapy, most of the conceptual problems of therapeutic evaluation occur in a diagnosis program even when decoupled from its eventual therapeutic partner. Therefore, we have concentrated first on understanding the patient description and diagnosis components of the above system outline and will be discussed in this paper. Our work on therapy and the problems of exactly how to interface diagnosis and therapy is in priliminary stages and will not be discussed in this paper.

1.3 Required Characteristics

In this section we discuss some of the characteristics required of the program if it is to be useful and effective as an expert consultant. These characteristics are:

1.3.1 Making the Proper Diagnosis

The primary responsibility of the program is to make the proper diagnosis. Without fulfilling this criterion, the program offers little possibility of being clinically useful. A criterion for deciding when a diagnosis has been achieved (and the program should move on to therapy) should weigh the costs of gathering further information in terms of morbidity, discomfort, time and money vs. the benefits of better diagnosis in terms of an improved management plan and a more reliable prognosis. For example, in situations when the management plan for each of the alternate diagnoses is the same, further diagnosis between alternatives does not have any utility from the point of view of formulating the management plan; hence, the diagnosis should be considered sufficient. Again, as the program is able to reevaluate the diagnosis as the disease evolves, it should be able to defer further diagnosis in favor of "creative indecision" [49, 50] in situations where the evolution of the patient's illness over time provides the best diagnostic clue at low risk.

1.3.2 Mode of Interaction

A distinction is often made between two forms of data acquisition in diagnosis: active and passive [15]. The active approach includes the computer's asking a question in order to solicit each new piece of information from the patient. A passive mode is one in which the program is provided with all the information at one point and must make a diagnosis based on this information. The active process suffers from the shortcoming that the physician may be aware of some facts potentially useful in the diagnosis, but may not be able to communicate them to the program because each new piece of information must be requested by the program. The passive approach avoids this problem but places the responsibility of identifying relevant information on the physician. But, as the user physician is not expected to be an expert in the medical domain of the program, this restriction is unacceptable. Therefore, for the purpose of this program we have chosen a variation of the active process, where the user is allowed to provide any amount of initial information (called presenting information), and from then on the program seeks information in the active mode. At any point in the diagnosis we allow the user to force the program into the passive mode and enter desired new information.

1.3.3 Diagnostic Style

The diagnostic style used by the program is almost as important as reaching the correct diagnosis. If the program jumps around among different medical problems under consideration, i.e., if the program's behaviour is erratic, the user cannot have a good model of the program's behaviour; makes the communication with the program confusing and annoying. If the program pursues some low priority diagnostic problem in the face of more important diagnostic problems of a life threatening nature or if the stream of questions seem endless, (i.e., if the program continues to ask questions when it should have been prescribing treatment or waiting for the patient to respond to treatment) it is likely to make the program practically useless.

We wish to design a program which will exhibit focused, coherent and purposeful behaviour in problem solving and will know when to stop any given diagnostic session.

1.3.4 Explanation

The conclusions reached by the program should be supportable by a medically meaningful explanation of how the conclusion was reached. Without the ability to explain its behaviour and knowledge, the program appears as a "black box", the theory of its operation is nearly impossible to examine critically or to correct, and its recommendations are likely to be greeted with scepticism by the medical community. Therefore, the communication of reasoning and knowledge used by the program is critical to the credibility and acceptability of the computer program as a clinical aid and also for its usefulness as a model for future development of the technology.

Although the program developed in this proposal does not have an explanation facility, it is designed to provide the basic capabilities needed for its implementation.

1.3.5 Handling Discrepant Information

One of the important problems in clinical medicine is the amount of discrepant information which must be dealt with. Some of the discrepancies arise because patients are not always accurate observers of their symptoms and because laboratory tests and medical records are often in error. In other cases the discrepancy may arise because of incomplete information, i.e. there may be a valid (but so far unknown) explanation for the discrepancy. Evaluation of a patient's condition in the face of discrepant information can penalize the ultimately correct hypothesis, thus misleading the program. In addition, there are many cases in which the discrepancy is not absolute, but relative to some currently believed hypothesis about the patient. The expectations of the physician play an important part in identifying and locally evaluating these discrepancies in the incoming information. This allows the physician to evaluate these discrepancies (with respect to the available evidence, physiological possibilities and the current hypothesis) and act upon them before assimilating the new information into his patient descriptions.

The importance of good handling of discrepant information can not be downplayed, especially when the system is expected to be used in a normal clinical setting as well as in experimental situations.

1.3.6 Continued Management of the Patient

Typically, a patient is examined by a physician more than once. Thus, the interaction between the patient and physician can be divided into the initial interaction and follow ups. The follow up sessions are used by physicians in evaluating their management plans and in refining the diagnosis. In most cases the follow up sessions are essential for arriving at the proper diagnosis and appropriate management of the patient, and the usefulness of a consultant is grossly inadequate if it cannot follow a given patient over a period of time.

1.4 Previous Work

In recent years there has been a growing interest in the use of computers as a means of providing expert advice on diagnosis and management of complex clinical problems. This has resulted in a large number of computer programs. A discussion of all of them is not possible. Therefore, I restrict myself to a few programs of direct interest to this proposal.

1.4.1 Acid Base and Electrolyte program of Bleich

The most widely known program for the management of electrolyte and acid base disorders was developed by Bleich using flowchart techniques [2]. This program takes as its starting point the electrolyte abnormalities of the patient and provides a profile of differential diagnosis and therapeutic recommendations. The effectiveness of this program is considerably reduced for the following reasons: it cannot deal with the clinical context in which the abnormalities are presented, evaluate the effects of therapeutic interventions and respond to an evolving clinical situations, and it cannot integrate abnormalities into coherent etiologic presentations.

Most of these problems are largely inherent to the flowchart approach. The flowchart programs are implemented by mapping out the exact order of questions for all possible responses, and by associating a diagnosis with each final branch. However, for a reasonably difficult medical situation where many alternatives must be considered, and where the interpretation of any finding depends on a large number of other findings, the complexity of a flowchart is inordinate. Further, a flowchart is not easily extendible because it involves determining the ideal place(s) for inserting a new question and updating the interpretation of the answers to every subsequent question. Since flowcharts describe only how the questions are asked and not why they are asked, they are not useful in explaining the medical implication or the reason for asking any question. Lastly, this technique is not adaptable to dealing with uncertain information or multiple initial complaints. A detailed discussion on this topic is given by Szolovits [49, 50].

The rich variety of data representation and problem-solving mechanisms developed in the field of artificial intelligence has attracted a number of projects in the medical area. The AI programming languages and programming philosophy provide the flexibility to to bypass some of the rigid limitations of the flowchart technique. In particular, in most AI schemes, the collection and application of rules is more modular and dynamically alterable (adaptable) than 'the flowchart approach and the availability of data representation methods allows us to represent the medical knowledge, the patient information, different interpretations of findings and alternate diagnoses and to use them in diagnosis and treatment. Some notable programs using AI techniques for medical diagnosis and therapy are: INTERNIST [30, 30, 33, 33], PIP [27, 28, 46], GLAUCOMA [53, 55] and MYCIN [6, 7, 42, 43].

1.4.2 Internist

Internist is by far the best known AIM program. It is based on a large data base and a relatively simple evaluation and problem-selection strategy. The Internist data base is constructed by linking diseases and their manifestations with two subjectively assessed scores; an "evocation" which describes how strongly the manifestation should suggest a disease, and a "frequency" which describes how commonly the particular manifestation is observed in a patient with the given disease. During the diagnostic process, all diseases with at least one reported manifestation are scored[1] (evaluated to yield a numerical estimate of relative likelihoods) and these disease hypotheses

[1] The score is computed on the basis of the following factors:

a. counting in favor of each hypothesis is a factor proportional to the combined evoking strengths of all observed manifestations explained by that hypothesis.
b. counting against a hypothesis are two factors: data not explained by the model, and data expected but found to be absent in the patient.
c. in certain cases, a "bonus" value is awarded; those models that are linked (either causally or by virtue of an equivalence relation) with disease nodes that have already been confirmed have their weights increased by an amount proportional to the degree of association between these links.

are partitioned into competing and complementary sets.[2] The competing set is further reduced by considering only those hypotheses whose weights are within a fixed range of the highest-scoring hypothesis and the next question is selected depending on the number and relative scores of the hypotheses in the competing set. The question selection strategies are; "confirm", to confirm the leading hypothesis in the competing set, "discriminate", to select a differentially significant question for the two leading competing hypotheses, or "rule-out", to eliminate the least likely of a large number of poorly discriminated hypotheses. INTERNIST also classifies all its diagnoses into a disease hierarchy. The disease hierarchy is a useful mechanism for controling the proliferation of active hypotheses during the diagnostic process because it allows the program to substitute a single general diagnosis for a class of diagnoses when no discriminating information is available to choose among them.

Internist represents medical knowledge as well as the patient specific facts in phenomenological terms (based on commonly observed co-occurrence and succession but without an understanding of the underlying mechanisims). The resulting lack of physiological knowledge affects the program in the following ways. The program cannot separate active hypotheses in competing and complementary classes effectively. For example, diseases such as Pyelonephritis and Urinary Tract Infection, which are generally competing can in some instances be complementary. The distinction between these two situations is essentially based on the physiological knowledge about the interactions between them. Similarly, if the simultaneous presence of two diseases alters their respective presentations significantly, the program is unable to confirm either one of them. These problems can only be overcome by either a deeper understanding of the disease mechanisms or the explicit representation of each such situation in the program. Further, the lack of physiological knowledge also results in activation of all phenomenologically possible hypotheses, although most of these possibilities are physiologically improbable or impossible. This increases the efforts needed in scoring and ruling-out these hypotheses.

The lumping together of findings and causally consequent diagnoses as manifestations leads Internist to further difficulties. For Internist, any manifestation is either present, absent or unobserved. This may be appropriate for findings, but when imposed on the evaluation of diagnoses, it ignores the arguable real support of a strongly suspected though not confirmed diagnosis for its causal antecedents. A similar deficiency arises because the findings are explained only by confirmed diagnoses. Hence a strongly suspected but unconfirmed complementary

[2] The partitioning algorithm can be described using following two concepts.

Shelf: a list of important manifestations that are not explained either by this diagnosis or any diagnoses previously confirmed.
Dominance relation: A hypothesis A is said to dominate hypothesis B if the net shelf of A (ignoring items explained by previously confirmed diagnoses) is a subset of the net shelf of B.

The competing set contains hypotheses that either dominate or are dominated by the highest-scoring hypothesis.

hypothesis will not be able to explain its significant findings, and so the correct diagnosis may have its scores strongly penalized for not explaining them. In addition, a given manifestation can be accounted for only once (by the first confirmed diagnosis that can explains the manifestation). Therefore, if a manifestation is caused by two different diseases, confirmation of one of them will prevent the other diagnosis from gaining support from it.

The notion of the patient-specific-model has not been well developed in Internist. The patient-specific-model in Internist consists of a collection of patient-facts and active-hypotheses; it does not relate different findings and hypotheses causally into diseases presentations. As a result the program has only a fragmented understanding about the patient's condition.

The problem solver in Internist alternates between generating a question (based on its hypothesis list) and generating and evaluating the hypothesis-list (based on user response to the question). Each question in the program is treated as an independent entity; the program does not group questions in clinically meaningful groups. Further, the program does not generate meaningful expectations about incoming information. This causes the question asking behaviour of the program to become erratic and vulnerable to discrepant information (see chapter 5).

Internist does not deal effectively with patients in whom the disease presentation is altered by prior treatment. Moreover, it does not include therapeutic considerations in its diagnosis and does not have any criterion to decide when a sufficiently complete diagnosis has been achieved. These deficiencies result in a program that continues to pursue diagnosis even after all the ailments have been identified.

Although some of the above criticisms also motivate the development of INTERNIST II, which should provide better answers than the original program, this research is still in a preliminary stage.

1.4.3 Present Illness Program

The Present Illness Program (PIP) is a frame based [23] program for taking the present illness in the domain of renal diseases. The PIP data base is implemented using disease frames, each containing the relation of the given disease to the findings and to other diseases, and a scoring criterion for evaluating the disease hypothesis. The findings associated with a disease are divided in two categories: "finding" and "trigger". The complementary relation between diseases is described using "causal", "complicational" and "associational" links; the competing relation is expressed using "differential" links. Each disease frame also contains two types of scoring functions; the "logical decision criteria" and the "numerical likelihood estimator" where the first is used for categorical evaluation and the second for probabilistic evaluation of the likelihood of the disease hypothesis under consideration [48].

During execution of the program, every disease frame is assigned a state as follows. A frame (disease hypothesis) is activated if one of its trigger finding is reported or if the frame is semi-active and any one of its findings is reported. A frame is semi-activated when any one of its causally associated frame is activated. The set of active frames in the program represents the disease hypotheses currently under consideration. Each active hypothesis is assigned a score which describes its likelihood in the given situation. A hypothesis is "confirmed" when its likelihood is sufficiently high or "inactivated" if it is sufficiently low. During the diagnostic process every active hypothesis is scored and the most likely one is chosen for confirmation. The program undertakes its confirmation procedure by asking about a (new) finding supporting it. The above process is repeated for every new finding entered in the program.

Next, we look at the differences between PIP and Internist in two areas: the data base and the techniques of problem solving. The PIP data base explicitly represents complementary and competing relations between diseases. This allows it to partition disease hypotheses in competing and complementary classes more efficiently than Internist. But, as these relations are static (do not depend on specific disease presentations in a patient) the program can not deal with situations where a pair of diseases can occur both as complementary and competing such as Pyelonephritis and Urinary Tract Infection discussed above. PIP treats the findings and diseases as different entities, avoiding the problems arising in Internist from treating them both as manifestations. Lastly, unlike Internist, PIP does not have a disease hierarchy. Therefore, it is unable to restrict the number of active hypotheses by substituting a single general diagnosis for a class of active hypotheses.

Similarly to Internist, the problem solver in PIP alternates between asking a question and generating and evaluating the hypothesis list, resulting in erratic question asking behaviour. Further, PIP has only one basic strategy, that of "confirmation" and does not use "differentiate" or "rule-out" strategies in problem solving. Therefore, the problem solver in PIP is somewhat less efficient than that of Internist.

Lastly, the PIP can deal with patients in whom the disease presentation is altered by prior treatment (it considers prior therapies as findings). But, it does not deal with recommending therapies or include therapeutic considerations in its diagnosis and does not have a criterion to decide when sufficient diagnosis has been achieved.

1.4.4 Glaucoma

The Glaucoma program deals with the diagnosis and treatment of eye diseases. In a production sense, it is the most developed of the four programs. It is implemented using the CASNET [53] theory of representation of causal knowledge. The medical knowledge in Glaucoma is represented as a network of physiological states. These states are linked together by subjectively assessed transition probabilities encoding the program's knowledge of how likely each causal link in the system may be and by support values indicating how strongly certain test results support the presence of a particular condition (state). The transitional probabilities are used primarily as a means of selecting the most appropriate next node to investigate and the support values are used to evaluate the score (fuzzy likelihood [13, 56]) of a state, which is used to confirm or deny it. Finally, the patterns of confirmed and denied nodes in the network are interpreted using a number of programs which, when initiated by the presence of an etiologically primary state, compare the progress of the diseases in the given patient with the diseases known to result from that primary disturbance.

The use of physiological knowledge gives the glaucoma program a better understanding of the mechanisms of disease evolution and interaction than the other programs discussed above. But the use of the physiological knowledge is restricted mainly to selection of the most appropriate next state for investigation. The program also lacks global diagnostic perspective because all the causal knowledge is described uniformly at a single level of detail; it cannot group its causal knowledge in terms of aggregate physiological conditions or syndromes and it separates the process of taking the present illness from that of diagnosis.[1] Moreover, the program works in a domain where the disease physiology is well understood and each state can be confirmed directly using some test. Therefore, the techniques developed in this program are not easily extendible to programs working in other domains of medical expertise.

1.4.5 Mycin

The Mycin is a rule-based program [7, 43] for diagnosis and treatment of infectious diseases -- in particular, bacterial infections in the blood (and recently extended to other infectious diseases). It represents medical knowledge in terms of production-rules [7] and uses a context tree to represent the patient specific knowledge [42, 43]. A novel mathematical model of confirmation [43] selects a set of organisms suspected of causing the illness. Diagnosis is carried out using a simple goal-directed control structure with backward chaining. The highest-level goal of Mycin is to determine if the patient is suffering from a significant infection which should be treated, and if he is, to select the appropriate therapy. It retrieves all the rules applicable to this goal and applies them sequentially as follows. It attempts to ascertain whether the "conclusion" of a rule is valid, by evaluating each of its premises. If this information is already available in the data base, the program retrieves it. If not, determination of this premise becomes the new objective, and the program recurses. If after trying all the relevant rules, the answer still has not been discovered, the program asks the user for the relevant clinical information which will permit it to establish the validity of the premise clause. Thus, the rules "unwind" to produce a succession of goals, and it is this attempt to achieve each goal that drives the consultation.

[1] The information gathering is directed towards confirming (ruling-out) pathophysiological states in the causal net. During this phase the program does not attempt to identify diseases responsible for the presence of these states (this is done after the information gathering phase is completed).

The rule-based Mycin methodology is applicable in fields where the domain specific knowledge can be described using judgmental rules. It appears to require a field which has attained a certain level of formalization with a generally recognized set of primitives and a minimal understanding of basic processes and which does not have a high level of interaction between conceptual primitives [6].

The rules in Mycin are used to represent the domain knowledge as well as to encode the flow of control of the program. This results in a somewhat unnatural representation of knowledge and it also takes away some of the advantages of modularity of knowledge (because one must take into account the possible interactions between rules during problem solving). The goal structure of Mycin allows efficient problem solving and can be used for explaining the problem solving behaviour of the program; but the program cannot explain the medical significance of its behaviour as this information is compiled out while writing the rules. Lastly, as the size of the domain increases the number and the complexity[1] of rules increases substantially.

1.5 Overview

In this paper we are dealing with diagnosis, i.e., the process of actively seeking information and identifying disease process(s) causing the patient's illness. In other words ascertaining *what the facts are* and *what the facts mean*. The process of active information gathering is dependent on our understanding and analysis of the available facts. From protocol analysis, researchers [8, 20, 32] have observed that doctors generally use a set of hypotheses (tentative diagnoses) to organize and search for new information efficiently. Therefore, it seems reasonable to organize the program around a problem solving paradigm using hypothesis generation mechanisms. The use of hypotheses will allow the program to organize sets of findings specific to a patient in a small number of chunks. It will also provide the program with a mechanism (commonly shared with the clinicians) to communicate its "thought process" to the clinician. All existing AIM programs discussed in this chapter use this parddigm.

From our experience with the existing diagnostic systems and a careful evaluation of the Present Illness Program, we are convinced that a relatively simple representation of what a physician knows about the patient (the physicians understanding about the patient's illness and the current state of the hypotheses about a patient's condition) will not be able to provide the program with the desired level of expertise. The patient description must unify all the known facts about the patient, their suspected interrelationships, the hypotheses and how the hypotheses account for various known and hypothesized findings. As the physician's knowledge is expressed at various levels of detail, from deep physiological and causal knowledge to global knowledge about syndromic and phenomenological correlations, we need a representation scheme that can unify these descriptions. We also need mechanisms for moving from one level of description to another. The patient description and associated knowledge representation which will be developed in chapter 2

[1] This is not necessarily true, but is evident from the evolution of MYCIN project.

attempts to construct a hierarchic description scheme with the above objectives in mind. It is the central patient specific data structure of the system.

Focusing our attention on the interaction between the hypothesis generation and the question selection process, let us assume for the time being that we have the capability of generating a small number of probable hypotheses given a set of findings. Now let us consider a program which alternates between asking a question and generating from scratch a new set of hypotheses after each new fact is entered. The regeneration of a diagnosis with every new fact is inherently inefficient, it requires a large amount of information processing (much of it repetitive) and it lacks the continuity of thought process so essential to human problem solving. This problem can be avoided by using a hypothesize and reformulate paradigm where a set of initial hypotheses is generated from the chief complaint and initial patient information; these hypotheses are then reformulated and extended to incorporate new findings acquired during the active information gathering process. In chapter 3 we discuss various data manipulation operations for creating and maintaining this structure. In chapter 4 we have extended the patient description to account for multiple hypothesized diagnoses arising from uncertainty and lack of knowledge about the patient.

Lastly, in chapter 5 we discuss the information acquisition and problem solving process. The task of diagnosis ideally involves confirming the appropriate hypothesis (or hypotheses) and eliminating other competing hypotheses. Note that multiple hypothesized diagnoses arise due to uncertainty and lack of knowledge about the patient. Further, the uncertainty is reflected in the patient description by states with a low belief-factor (score) and the lack of knowledge is reflected by multiple competing interpretations. Now the task of the diagnostic problem solver can be reformulated as that of identifying and eliminating the uncertainty and lack of information in the patient description. This is done as follows. All the states (in the patient description) with low belief factors and multiple competing interpretations are collected, forming a problem-set. This problem set is used to formulate the top level diagnostic goal. A plan for problem solving is generated by decomposing this goal successively into subgoals in the context of the patient description. Each goal in the plan is associated with the parts of the patient description relevant to it and the prior expectations about the outcome of the problem solving effort. The association between the goal structure and the patient description allows us to separate patient specific information relevant to the immediate diagnostic problem from information not directly relevant. The expectations associated with the goal structure provides us with the context in which to evaluate the incoming information for discrepancies.

The decomposition of the diagnostic goal into subgoals is done using various diagnostic strategies based on the protocol analyses by Miller [21], Kassirer [19] and Elstein [9]. The strategies are Confirm, Rule-Out, Differentiate, Group-and-Differentiate, Refine and Explore. The choice of an appropriate strategy for a given situation is based on the number of hypotheses in the primary-goal and their relative belief values.

2. Representation of Medical Knowledge

The knowledge used by a diagnostic program can be divided into two classes: knowledge about diseases and their presentation in a patient, and heuristic problem solving knowledge. In this section we will discuss the representation of medical and patient specific knowledge and develop a patient specific model. As the two representations share general descriptive schemes, we will deal with them concurrently. The representation being discussed will be specifically indicated if it is not clear from the context.

Illness can be described as a change in the normal state or function in a patient. To describe an illness, we need a formalism to represent the states, the state changes, the normal and the abnormal functions and their interactions in a patient in terms of the primitives known to the system. It is also important to recognize various composite situations in order to get a global perspective of the patient's illness. Recognition of these situations or compound descriptions in a diagnostic system is important because it provides us with the ability to reason at a high level of abstraction, organizing a large number of seemingly unrelated facts; more importantly, we may use clinical and diagnostic knowledge which is generally organized around high level concepts. Therefore, the diagnostic system should allow descriptions to be in terms of both the high level concepts such as diseases and the detailed physiological states and processes underlying the presentation of the illness in a patient.

2.1 State

Following Forrester, we say that the states in the system describe two kinds of variables: levels and rates [12]. The levels are the accumulations within the system. They represent the present values of those variables that have resulted from the accumulated differences between the inflows and outflows. Rates define the present, instantaneous flows between the levels in the system. The rates correspond to activity, while the levels measure the resulting state to which the system has been brought by the activity. A state in the system is described by its temporal characteristics, severity and other aspects relevant to the state. A state is a "primitive-state" if it does not contain internal structure and is a "composite-state" if it can be defined in terms of other states.

One important function of diagnostic reasoning is to relate causally the diseases and symptoms observed in a patient. These causal relations play a central role in identifying clusters that can be meaningfully aggregated and in developing coherent diagnoses. The presence or absence of a causal relation between a pair of states can change their diagnostic, prognostic and etiologic interpretations. Therefore, the system should have the capability of hypothesizing about the presence or absence of a causal link. We also note that, in a manner similar to state descriptions, a causal relation can be described at various levels of abstraction.

2.2 Link

A link specifies the relation between two states. In the past (PIP, INTERNIST, GLAUCOMA), links were used to describe causal relations between states. From our study, we have come to the conclusion that this single representation of the interaction between states is inadequate, because this representation forces us to assume that every interaction between states is causal (or statistical) in nature. We believe that the interaction between states occurs at various qualitatively distinct levels. For example, two states may be causally related to one another, or they may be associated with one another in a statistical sense, without any known causal relation between them, or the presence of one of the states may alter the interpretation of the other state without changing the likelihood of the other state in any significant, predictable way. To capture these differences, we will use three types of links described below. Further, we believe that a link between a pair of states in the medical data base may or may not be present in a given patient (for example, although hypotension and acute tubular necrosis are causally related in general, they may not be related in a given patient.). Therefore, in order to reason with relations between a pair of states in the patient specific model, we must instantiate links and incorporate them in the model.

2.2.1 Causal link

A causal link specifies the "cause-effect" relation between the "cause" (the antecedent) and the "effect" (the consequent) states. In the past (PIP, INTERNIST), causal links were described by specifying the type of causality (may-be-caused-by, complication-of, etc.), and a number representing in some form the likelihood (conditional probability) of observing the effect given the cause. From our study, we have noted that the conditional probability of observing an effect given the cause or vice versa, depends upon various aspects of the cause, such as severity, duration etc. as well as other factors in the context in which the link is invoked (such as the age, sex, weight, etc. and the current hypothesis about the patient). For the effective use of a causal relation, we need to take these conditions into consideration. To illustrate this, let us consider (a simplified) causal relation between diarrhea and metabolic-acidosis. In terms of conditional likelihood, we could state the relation as P(Metabolic-Acidosis/Diarrhea) = 0.7 or the probability of observing Metabolic-Acidosis given Diarrhea is 0.7. A rule-based description of the causal relation can be specified as follows;

```
IF DIARRHEA IS SEVERE AND ITS DURATION IS GREATER THAN TWO DAYS
   THEN
     IF THE PATIENT HAS NOT RECEIVED BICARBONATE-THERAPY RECENTLY
        THEN THE PATIENT MAY HAVE MODERATELY SEVERE METABOLIC-ACIDOSIS WITH NORMAL ANION-GAP
        ELSE THE PATIENT MAY HAVE A MILD METABOLIC-ACIDOSIS WITH NORMAL ANION-GAP.
```

From the above example it is apparent that the conditional probability of observing metabolic-acidosis and its severity and duration depend on the severity and duration of diarrhea and bicarbonate-therapy. Generalizing, it appears reasonable to expect that causal links between the cause and effect nodes should contain the information on how an instance of the cause relates to an instance of effect and other factors influencing the relation.

A causal link in the system is an object denoting the causal relation between a cause-effect pair. It specifies a multivariate relation between various aspects of the cause and the effect, taking into account the context and the assumptions under which the causal relation is being instantiated. A schematic description of a causal link is presented in figure 2.2.

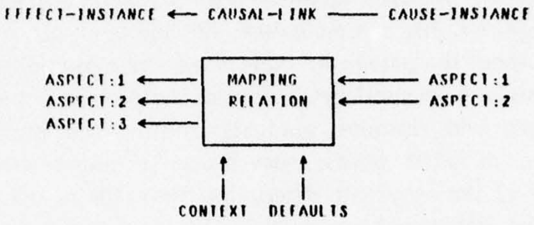


Fig. 2.2

2.2.2 Associational links

This link states that the presence of one state influences the expectation about the presence or absence of the other state. It suggests that the two states are correlated, but does not specify the reason for the correlation or association between the states. For example:

HYPONATREMIA — ASSOCIATED-WITH ⟶ CENTRAL-NERVOUS-SYSTEM-SYMPTOMS

Fig. 2.3

it is well known that severe hyponatremia is associated with central nervous system disorders such as coma, stupor and confusion. But the specific physiology for the causation of these disorders is not well understood.

2.2.3 Grouping Links

Grouping links state that the presence of two states simultaneously (in the patient specific model) represents a situation which is recognized as a part of some useful abstraction. This link does not imply any correlation or causal connection between the states connected by the link. In other words this link is used to group together states without any commitment to their mutual cause-effect relation. For example, let us consider the following group of symptoms, severe hyponatremia, low creatinine (less than 0.6) and normal bicarbonate.

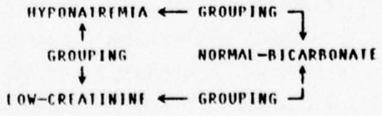


Fig. 2.4

This group of findings is strongly suggestive of SIADH (Inappropriate secretion of antidiuretic hormone). They are grouped together in the program, which can then be used to suggest SIADH at higher levels of aggregation.

2.3 Hierarchic Description of States and Link

In this section we will discuss a hierarchic representation scheme for the description of the patient specific knowledge. In this representation, the low end of the hierarchy describes the physiological knowledge about the patient in terms of primitive physiological concepts and relations known to the system. This physiological description of the patient is then successively aggregated into higher level concepts and relations, gradually shifting the emphasis from physiologic to syndromic description or, in other words, from causal to phenomenological descriptions. The phenomenological nature of the aggregate description allows us to use efficiently the hypothesize and test paradigm [24] for global problem solving. The causal nature of the detailed description allows us to use causal reasoning and to restrict the number of hypotheses generated to a small number by imposing causal and physiological consistency requirements. We will first introduce the state and link aggregations with the help of examples.

2.3.1 Hierarchic description of states

To illustrate the concept of a state aggregation, let us consider the condition of excessive loss of lower gastrointestinal fluid (Lower-GI-Loss). The compositions of the lower-GI-fluid and the plasma fluid are as follows.[1]

	LOWER GI FLUID	PLASMA FLUID	
Na	100 - 110	138 - 145	mEq/l
K	30 - 40	4 - 5	mEq/l
Cl	60 - 90	100 - 110	mEq/l
HCO3	30 - 60	24 - 28	mEq/l

Fig. 2.5

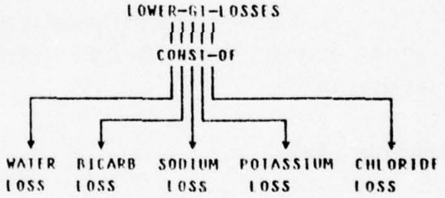


Fig. 2.6

In comparison with plasma fluid, the lower GI fluid is rich in bicarbonate and potassium and is deficient in sodium and chloride (except in case of hyperchloremic diarrhea). This information is represented in the knowledge base by decomposing lower-GI-fluid into its constituents (and associating appropriate quantitative information with the decomposition) as shown in fig. 2.5. Now the loss of lower-GI-fluid (lower-gi-losses) would result in the loss of corresponding quantities of its constituents (in relation to the total quantity of fluid lost) as shown in figure 2.6 (the quantitative relation is not shown for reasons of clarity). Therefore, an excessive loss of lower GI fluid without proper compensation of electrolytes will result in hypobicarbonatemia, hypokalemia, hyperchloremia,

[1] In this section the following abbreviations are used. CBY (caused by), SBY (succeeded by or followed by), CONST-OF (constituent of), and COMP-OF (component of).

hypernatremia and volume-loss as shown in fig. 2.7.

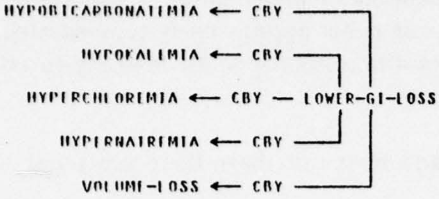


Fig. 2.7

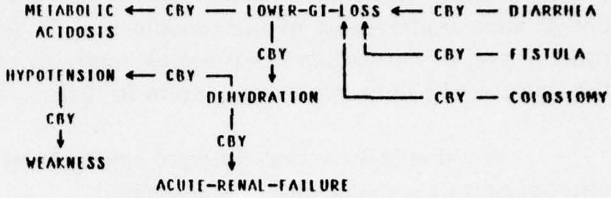


Fig. 2.8

At the next level of detail we can describe the causes and consequences of lower GI loss as shown in Fig. 2.8. This causal cluster can be further summarized to the description shown in fig. 2.9.

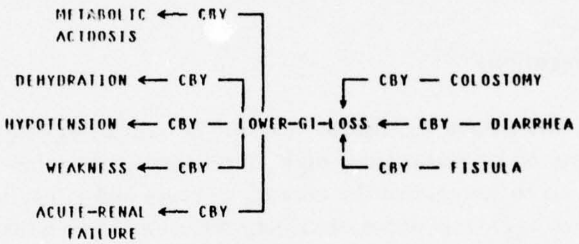


Fig. 2.9

The summarization of the description in figure 2.8 into that of figure 2.9 is performed using link aggregations described in the next subsection.

2.3.2 Hierarchic description of causal relations

To illustrate the concept of a causal link abstraction, let us consider the following aggregate causal assertion.

DEHYDRATION ← CBY — DIARRHEA

Fig. 2.10

The causal mechanism of diarrheal-dehydration can be explained as follows; diarrhea causes lower gastrointestinal fluid loss (Lower-Gi-Loss) which causes dehydration.

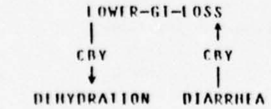


Fig. 2.11

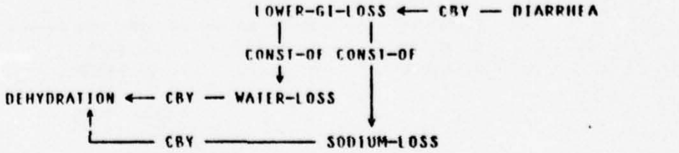

LOWER-GI-LOSS ← CBY — DIARRHEA
CONST-OF
CONST-OF
DEHYDRATION ← CBY — WATER-LOSS
CBY
SODIUM-LOSS

Fig. 2.12

In greater detail, the lower gastrointestinal fluid loss consists of water and sodium loss (along with other electrolytes not included here for the simplicity of presentation) which results in dehydration under either of two conditions: a) the water loss is not appropriately compensated by fluid intake; b) the sodium loss is not appropriately compensated, resulting in an inability to retain the water in the fluid intake as shown in fig. 2.12.

We should note that different aggregate states and links can share their sub-states with other aggregate states or links. For example,

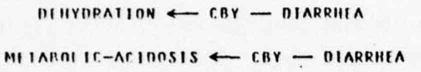

DEHYDRATION ← CBY — DIARRHEA
METABOLIC-ACIDOSIS ← CBY — DIARRHEA

FIG. 2.13(A)

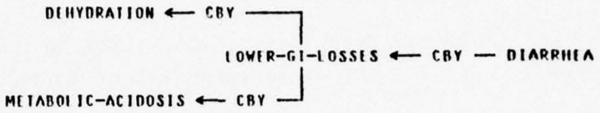


FIG. 2.14(B)

2.4 Types of Aggregations

The hierarchic patient-description structure is built by aggregating low level concepts into higher level concepts or by elaborating high level concepts into their constituents at lower levels. Aggregation allows us to summarize the network of cause and effect nodes describing the patient's illness by substituting aggregate nodes describing them for large chunks of the causal networks and by retaining only the important nodes (analogous to landmarks in a conceptual map [20]). The aggregations and elaborations can be divided into five classes, described below.

2.4.1 Temporal Aggregation

An observation provides us (clinician or the program) with a snap-shot of the state of some process. During the course of evolution of a patient's illness, a sequence of snap-shots becomes available to the program. These snap-shots describe the progress of a process through time. Temporal aggregation allows us to resynthesize the process and describe it as a single concept in the patient specific model. Further, temporal aggregation allows us to summarize a sequence of values of a state variable into a single qualitative description and allows us to make hypotheses about temporal patterns or sequences of values we expect to observe. For example:

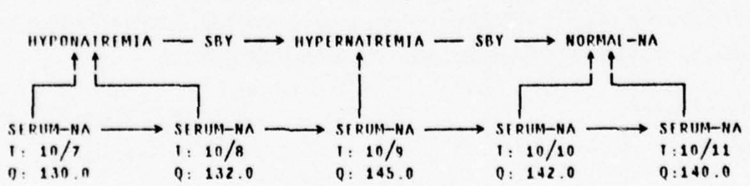


Fig. 2.15

2.4.2 Component Summation

Component summation becomes necessary in our program because the causal links in the system contain specific quantitative information on how the various aspects of cause and effect instances (in a given patient) relate to one another. Therefore, the matching of the cause-effect relation will be unsuccessful or undesirably affected if the aspects of the effect instance are distorted by other nodes. In order to prevent this from happening we must instantiate the effect of each cause exclusively and then combine them together. This is illustrated by the following example,

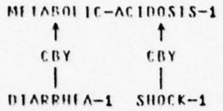


Fig. 2.16(a)

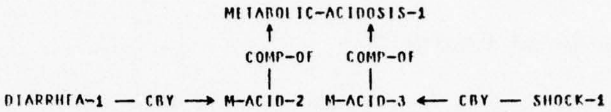


Fig. 2.16(b)

The above description is a superimposition of two relations: METABOLIC-ACIDOSIS-1 caused by DIARRHEA-1 and METABOLIC-ACIDOSIS-1 caused by SHOCK-1. The description does not explicitly state that the metabolic-acidosis (in the patient specific model) is partly caused by diarrhea and partly by shock. Therefore while considering the first relation, the program will assume that the *entire* metabolic-acidosis is being caused by diarrhea. This will result in the cause-effect relation being mismatched (because the severity of acidosis and anion-gap associated with it are larger than can possibly be attributed to diarrhea). A similar mismatch would also occur while considering the relation between the shock and the metabolic-acidosis. To avoid this problem let us *consider the description of the example as shown in fig.* 2.16(b). The state description of fig 2.16(a) can be considered as an abstraction of fig 2.16(b), where METABOLIC-ACIDOSIS-2 and METABOLIC-ACIDOSIS-3 are respectively the components of METABOLIC-ACIDOSIS-1 caused by DIARRHEA-1 and SHOCK-1. The decomposition of an effect with multiple causes into its causal components provides us with valuable information in evaluating prognosis and in formulating therapeutic interventions.

Taking another example, metabolic-acidosis could be considered to be hypobicarbonatemia causing a reduction in pH, which causes hyperventilation and reduced pCO2 which in turn causes an increase in pH, an example of negative feedback. The increase is less than the initial reduction, causing a net reduction in pH.

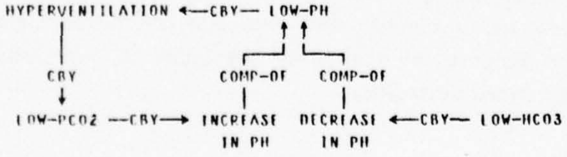


Fig. 2.17

The component summation mechanism allows us to separate the primary component of the change from the secondary feedback components in feedback loops and allows us to fold causal chains in feedback loops that represent continuous processes. On the other hand, decomposition of a state into its contributing factors raises a new problem, that of combining the factors together. The combined effect of two components may not be additive and may depend on their causes and on the physiological mechanisms involved in the particular case. This problem needs further in-depth study, but for the time being let us assume that there is some (in all likelihood a local) mechanism that allows us to combine the contributing factors together satisfactorily.

2.4.3 Constituent Aggregation

Many diseases and conditions have some internal physiological structure, or a set of physiological conditions that can be considered as definitional for the disease or condition. This knowledge is represented with the help of constituent-aggregation or disaggregation. An example of constituent aggregation is shown in fig. 2.6.

2.4.4 Causal Aggregation

Some of the possible groups of states are conceptualized and have names describing them as conceptual objects. In some sense, these concepts provide the structure for organizing different situations that are commonly encountered during diagnosis. Because of historic evolution in the recognition of situations and their nonstandard use, terms such as disease, syndrome, physiological states etc. have overlapping definitions. But the existence of these different terms signifies the existence of different types of clusters. We will try to identify the basic structure of these clusters without differentiating between the specific mechanisms responsible for their differences, mostly because we do not have sufficient knowledge in the problem solving domain to exploit the subtle differences between them. An example of causal aggregation is shown in fig. 2.8 .

2.4.5 Link Aggregation

Links are used to represent causal relations between states. The causal relations are understood and can be described at different degrees of specificity and detail. This has been illustrated in example 2.2. The link aggregation hierarchy represents an alternate hierarchy to the state aggregation hierarchy, and provides us with a different view point in summarizing the patient description by allowing us to identify and eliminate intermediate states from causal chains of reasoning. They also help us in evaluating the belief in the causal link at higher level of aggregation using their detail descriptions.

2.5 Summary

In this section we have developed a hierarchic description of a patient's illness based on our observation that disease states and their inter-relationships can be described at various levels of detail. The low end of this hierarchy describes the detailed physiological knowledge about the patient, which is successively aggregated into higher level concepts and relations, gradually shifting the emphasis from physiological to syndromic descriptions or, in other words, from causal to phenomenological descriptions. This hierarchy is built using temporal, component, constituent, causal and link aggregations. The aggregation process allows us to summarize the patient facts providing us a concise global perspective. As the physiological reasoning tends to be categorical and syndromic reasoning probabilistic, the hierarchic description allows us to blend together the use of categorical and probabilistic reasoning [48].

We should note that the patient description discussed above was developed with the assumption that the true state of the patient is known. In the next chapter we will extend this description to take uncertainty into account.

3. Representation of Uncertainty

In previous sections we have discussed the representation for describing illness in a patient assuming that the true state of the patient's illness is known. But in reality the knowledge about the patient is incomplete and uncertain. Its incompleteness gives rise to more than one possible (hypothesized) diagnosis for the patient's illness. In this section we will study two major ways in which this uncertainty appears and will extend the patient description using the concept of hypothesized description to account for the uncertainty and the possibility of more than one hypothesized diagnosis.

Uncertainty in the patient description appears in two related but distinct ways. First, because of incompleteness in our knowledge about the true patient description (ignorance) and second, because of the inherent chance nature of the disease process. To illustrate the differences between the two, let us consider a patient with severe diarrhea and vomiting. The above findings are sufficient to assume that the patient may have metabolic disturbance, but not enough to specify the type of disturbance (which depends upon the relative amounts of fluid lost due to diarrhea and vomiting). This can be represented by specifying a high level of belief for metabolic disturbance and not specifying any belief value for metabolic acidosis or metabolic alkalosis, thus preventing the program from making any inference based upon the relative likelihoods of metabolic acidosis and alkalosis. Thus we note that a given level of ignorance of the program can be represented by asserting concepts at an appropriate level in the classification hierarchy.

On the other hand there are many situations where our medical knowledge about a disease process is sufficiently incomplete that no deterministic theory exists; any consequences we derive, then, must be inherently of a chance nature. For example, only a small fraction of people suffering from streptococcal infection develop acute glomerulonephritis. The uncertainty in these assertion arises because of the probabilistic nature of *medical knowledge* and not because of a lack of information about the *patient's illness*. The chance nature of such an assertion e.g., the likelihood that a given hypothesis is true, must be described using a probabilistic measure. In case the simple characterization of our belief in a hypothesis by means of the probability is not sufficient, we could extend the probabilistic measure to a triple consisting of the minimum degree of support, the expected support and the maximum possible support [41, 47].

The problem of evaluating the measure of belief in a given hypothesis is a difficult one. Various people have worked on this problem [8, 41, 42, 48, 56], but no good practical solution has been found because the simplifying assumptions made in the most often used probability theory (e.g., those of completeness of the universe of discourse, conditional independence of events etc.) are not valid here. In addition the issues of the incompleteness and uncertainty of information have often been confused. We wish to separate these two issues in order to exploit the capabilities of AI techniques, which are much better at handling incomplete information (by hierarchic description) than dealing with uncertainty. This would allow us to reduce our dependence on less than adequate schemes available for dealing with uncertainty.

3.1 Hypothesis

A hypothesis is a supposition or conjecture put forth to account for known facts [52]. Stated differently, a hypothesis can be considered to be a meta level concept describing a relation between two objects and specifying the type of relation (being hypothesized between the two objects) and our belief in this relation. To illustrate the various types of relations possible, let us consider the following three hypothesized relations:

Type-1: the patient has hyponatremia,

Type-2: the hypotension of the patient is caused by hyponatremia, and

Type-3: the symptoms observed correspond to acute glomerulonephritis.

The type-1 hypothesis expresses our belief in a particular state being the true state in the patient. It is used to describe the degree of belief in a state-description (or diagnosis) at any *level of* abstraction. The type-2 hypothesis describes the degree of belief in a link between two states. Note that the degree of belief in the link is independent of our belief in the states associated with the link. Therefore, it is possible to construct a situation where the belief in the link is high while the belief in the states being linked may be quite low. This would correspond to holding the point of view that neither of the two states are likely, but that if they are the true states, then it is very likely that they are related by the link represented in the hypothesis. The type-3 hypothesis expresses our belief in the appropriateness of an abstraction. In other words, it expresses our belief in the matching between a state-description and its abstraction, i.e. diagnosis. From type-1 and type-3 hypotheses it becomes apparent that two different belief values can be associated with a given diagnosis: these are, our belief in the appropriateness of a diagnosis given the patient facts and our belief in the patient facts.

In this section we have developed a descriptive mechanism to represent a hypothesis and uncertainty about the hypothesis. Ignorance about the patient facts is represented structurally by selecting the level at which a hypothesis is made. Uncertainty about the hypothesis is described explicitly by associating a belief measure with the hypothesis. In the next section we will extend this to the patient description structure in generating a set of competing hypotheses to explain the patient's illness.

3.2 Complete Hypothesis

At any point in the diagnosis the clinician (program) has a partial understanding of the illness of the patient. This understanding can be described by a set of hypothesized diagnoses, where each hypothesized diagnosis accounts for all the observed manifestations of the illness. In the program, a diagnosis is represented by a patient description structure (described in Section 2). Each such patient description structure with its associated hypothesis structure is called a "complete hypothesis" and abbreviated as "CH". Note that each complete hypothesis provides an alternate explanation for the disease process in the patient and only one of these complete hypotheses can be correct. Thus we have a set of alternate diagnoses which are mutually exclusive and competing; therefore, they can be rank-ordered according to their likelihood.

At any point in the diagnosis there are only a few significantly distinct explanations for the patient's illness, although each such explanation can have a substantial number of variations. This problem can be contained by selecting an appropriate level of detail for representing the patient description. This should allow us to represent small differences in the hypotheses implicitly (by professing ignorance about them) while focusing on the major differences. Again, during problem solving we will need to compare the different alternatives (complete hypotheses) to identify the differences between them. If each complete hypothesis is represented separately, this task can become substantial. This problem can be overcome by allowing different hypotheses to share common sub-hypothesis structures, thus producing a single structure to represent the set of alternate CH's in which the important differences move up the structure, while the smaller differences tend to be buried deep inside the structure.

3.3 Summary

In this chapter we have noted two important causes of uncertainty about the patient description; the chance nature of the disease process and lack of information about the disease manifestation in the patient. Next we have developed a mechanism for associating uncertainty with the patient description, allowing us to represent hypotheses about a state-description, causal links between state-descriptions and matching between state-descriptions and diseases. Lastly, we have developed the notion of a set of competing hypotheses where each complete hypothesis is a consistent description of a patient's illness that can account for all the observed findings. We have also noted that each complete hypothesis represents a diagnosis and competes with other complete hypotheses to explain the patient's illness. Further as these complete hypotheses are mutually exclusive and competing, they can be rank ordered as a set of alternate diagnoses.

4. Description Building Operations

During the design of the program, we have tried to separate the data structures required for the representation of a patient description and those required for efficient problem solving. This allows us to represent the patient description, incorporating all the available findings and derived facts about the patient with the objective of semantic clarity, coherence, completeness and explainability. In the preceding section we have studied the structural organization of the patient specific model. In this section we will outline the operations used to create and maintain the patient specific model as new information is added to it.

In this section we will study the operations for creating and augmenting the patient description. These operations are INITIAL-FORMULATION to create an initial patient description from the presenting complaints, initial findings and lab-results; AGGREGATION to combine various findings into causal clusters representing different disease hypotheses (moving up); ELABORATION to decompose aggregate findings and hypotheses into their components or specific subclasses (moving down); and PROJECTION to hypothesize associated findings and diseases suggested by states in the patient description at the same level of abstraction (moving sideways).

4.1 Initial Formulation

From the observation of the clinical behavior of doctors, we have noticed that the initial response of a physician to the chief complaints is predetermined and depends upon the "clinical situation". On further pursuing this observation, we noted that the number of initial situations (i.e., the way in which the patient is initially presented to the physician) is not large. For example, in the domain of acid-base disturbances there are about 30 such situations. Therefore, it is feasible to compile a set of initial patient descriptions with situation specific information such as expectations, causes, consequences, probable diagnosis and initial set of exploratory questions (similar to playing the initial chess game using book moves). This will allow the program to set up a specific framework around which the incoming information can be organized and thus avoid a serious difficulty encountered by programs using hypothesize and test paradigm. That is, when there is very little information or when the available information is nonspecific, a large number of possible hypotheses are activated (for example, in PIP, the first few facts entered could trigger as many as half of all the possible hypotheses).

4.2 Aggregation

The aggregation operation is used to combine causally related states into clusters representing states at a higher level of abstraction as follows. When a set of new findings is entered in the program, it groups these findings into causally related clusters about which it is fairly certain. Next, it tries to group these clusters in alternate ways. If the number of alternate groupings is small (possibly two or three) the program builds alternate structures describing these possibilities. If the number of possibilities is large, the program tries to abstract already formed clusters into semantically larger concepts and tries again. This process is continued until it reaches a level of abstraction where most of the objects are connected to one another or when most of the objects are etiologies or diseases, at which point structural abstraction is no longer useful. Each abstraction generated by this process provides us with a coherent partial diagnosis for the patient. Within each abstraction pyramid all the diseases, findings etc. are mutually complementary, while the alternate abstraction pyramids provide us with competing diagnoses.

4.3 Elaboration

The process of abstraction in the system is complemented by the process of elaboration. When the program is provided with information about some disease or it generates some hypothesis at a high level of abstraction, it must assimilate this information in the patient description. This can only be done if sufficient structure exists at lower levels of hierarchy to structurally support this hypothesis. If not, the supporting structure must be constructed by decomposing this hypothesis into a more detailed presentation. Here again, we are faced with the situation that there may be many possible elaborations (presentations) for the same abstract condition. But as this hypothesis must be consistent with already known facts about the patient, parts of the elaboration will already be present in the patient model and the rest of the elaboration must be consistent with the model. This should reduce the number of possible elaborations greatly. Even then, the lack of information at lower levels of detail may cause a potentially large number of alternate elaborations to be possible and thus prevent us from elaborating any abstract hypothesis to the lowest level of detail without further discriminating information.

4.4 Projection

So far we have studied the data operations that allow us to move up (abstraction) or down (elaboration) to build the hierarchic description of the patient model. Here we will discuss the third major type of data operation that will allow us to broaden the cross-section of the model at any given level of abstraction. The basic process of projection can be described as follows. Suppose at any given instance we are considering some hypothesis H. If H is present then some of its antecedents (causes) or consequents (effects) must be present. Therefore, if hypothesis H is assumed, it is reasonable to assume that at least one of its antecedents and most of its consequents will also be present in the patient. The process of projection allows us to suggest new hypotheses about the causes and the consequences of a given node at the same level of abstraction. These new hypotheses can then be used to group different nodes into causally antecedent-consequent pairs allowing further abstraction or elaboration. Note that abstraction and elaboration operations do not suggest new hypotheses.

4.5 Summary

In this chapter we have discussed operations for creating and maintaining the patient description. The initial patient description is formulated by selecting one of the precompiled initial situations with the help of chief complaint(s), lab results and initial findings and asking questions associated with it. The use of a predetermined set of initial situations and questioning sequences allows us to avoid the inherent unfocused behaviour of the hypothesize and test paradigm due to its lack of information. Further, it allows us to provide the program a skeletal structure for incorporating and organizing incoming information in the early phase of diagnosis.

The patient-description is augmented using aggregation, elaboration and projection operations. The aggregation operation is used to combine causally related states into clusters representing states at a higher level of abstraction. Thus aggregation allows us to extend the patient hierarchy upwards. The elaboration operation is used to extend the patient hierarchy downwards by decomposing aggregate states into their causal components. It allows us to build a physiological description of the patient from disease descriptions. Lastly, the projection operation is used to hypothesize associated findings and diseases suggested by states in the patient description at the same level of abstraction. Thus it allows us to broaden the cross-section of the patient description hierarchy.

5. Diagnostic Problem Solving

The patient description developed above was designed to provide the program with the capability of expressing its understanding about the patient's illness. In the patient description we are interested in assimilating all the available information in a coherent form. On the other hand, for any specific diagnostic problem, a large portion of this information is not directly relevant. A diagnostic problem statement should focus on that part of the patient description where the understanding is uncertain. It should describe different alternatives (which can be differentiated), and it should be easily decomposable into smaller problems. In this section we will discuss the process of identification and formulation of the diagnostic problem, its representation and decomposition into subproblems, the problem solving strategies, and the use of expectations in identifying discrepant information and in directing the flow of control.

5.1 Problem Identification and Formulation

At any point during the diagnosis the program has a set of complete hypotheses vying to explain the patient's illness. The reason for this is the lack of discriminatory information needed to resolve the differences between the CH's. The addition of this information to the patient description should result in resolution of these differences. The specific places where this information is lacking can be identified by identifying places where two or more hypotheses differ from each other in interpreting the known findings. Each place so identified represents a potential diagnostic problem. All the diagnostic problems identified above are collected in a list (problem-set) and are used *in problem formulation.*

Viewed differently, the problem-set identified above describes the set of problems all of which need to be solved in order to differentiate between the competing Complete Hypotheses. The availability of a set of problems to work on simultaneously provides the problem solver with the ability to minimize the sum total effort needed in solving all the problems by abstracting common aspects of problems and by selecting an efficient order in which the problems are solved. This can be done using either a rank-ordering heuristic or a problem-abstraction heuristic.

i. Rank-Ordering Heuristic: This represents a first cut heuristic. Here the problem set is rank ordered according to some criterion such as the need and urgency of diagnosis, the therapeutic importance, global usefulness in patient understanding etc. and the problem with the maximum score is selected for problem solving.

ii. Problem-Abstraction Heuristic: Quite often, many of the problems in the problem set share some common feature, such as the duration and severity of the illness, the specific organ system or the disease mechanism involved. In such cases it is useful for the problem solver to abstract the common aspect of the problems or to partition the problem set in different classes and then differentiate between them, thus attacking a group of alternatives simultaneously. In short the problem formulator either formulates or selects one of the problems from the problem set for further problem solving.

5.2 Problem Description and Goal Structure

The problem formulated above is set up as the top level diagnostic goal for the diagnostic problem solver. A diagnostic goal consists of the following components; a) a primary goal which describes the main problem to be solved by the problem solver, b) a context which describes the reason for solving the problem and c) an expectation which describes the programs prior expectation about the outcome of the problem solving activity based on the knowledge already available to the program. These expectations are used in determining the consistency of the incoming information with the patient description and in directing the flow of control [21]. An example of a goal statement is given in the figure 5.20.

Once the top level diagnostic goal is identified the problem solver sets up a goal structure (a plan for problem solving) by decomposing this goal into subgoals recursively until we reach subgoals that can be solved using primitives known to the system. The subgoal generation is accomplished using elaboration and projection operations (described in Section 4) in the context of the patient description. Each subgoal is associated with some part of the patient description relevant to the the problem being decomposed. Therefore, we can view the goal structure as a representation of problem specific information extracted from the patient description. This interaction between the two allows us to assimilate the information gathered during problem solving with the patient description efficiently (as the interpretation and context in which the information is relevant to the patient model is known a priori). It also allows us to associate semantically meaningful expectations with goal statements to check the incoming information with the patient description for apparent and real contradictions.

5.3 Problem Solving Strategies

For a long time one type of strategy has dominated the thinking of the medical profession -- the differential diagnosis. The codification of this approach in a book such as French's "Index of Differential Diagnosis" was considered an important step forward in the systematic organization of diagnostic procedures. On the other hand, most of the computer programs using the hypothesize and test paradigm for diagnosis have generally emphasized the confirmation strategy confirmation is an important strategy in itself, its effectiveness is limited to situations where only one hypothesis is considered or one hypothesis dominates the competing hypotheses. In a study of strategy selection in medical diagnosis using protocol analysis, Miller [21] has identified various different strategies and situations where these strategies can be used for efficient diagnostic problem solving. In this program we propose to extend strategies identified by Miller using our experience with Internist [30, 31, 32, 33] and PIP [27, 28, 46] and adapt them for use in conjunction with the patient description developed in previous sections. These strategies can be broadly classified as follows: Confirm, Rule-Out, Differentiate, Group-and-Differentiate, Refine and Explore. The selection of an appropriate strategy is based upon the syntactic structure of the diagnostic problem (e.g., the number of alternate hypotheses being considered and their beliefs relative to one another) as described below. The confirmation strategy is used when we have only one hypothesis under consideration, or when among a group of hypotheses, one hypothesis is much more likely than all other alternatives under consideration. The rule-out strategy is used to eliminate some hypothesis, which is substantially less likely than all other alternatives under consideration. The differentiation strategy is used to discriminate between two hypotheses with similar belief factors. The group-and-differentiate strategy is used when we have a large number of alternate hypotheses with similar belief factors. Here we need to discard a large number of hypotheses rapidly in order to focus our attention on a small number of alternatives. This is done by partitioning the alternatives into a small number of groups according to some common characterization and then applying a differentiation strategy to rule-out (or confirm) one of the groups, thus narrowing the hypothesis set. The refinement strategy is used to refine a hypothesis about a general class of diseases into more specific hypothesis. Note that the refinement of a hypothesis into more specific hypotheses generally results in a disjunctive set of hypotheses. Therefore, the refinement strategy is generally followed by differentiation. Finally, the explore strategy is used when the patient description does not have any well defined diagnostic problems to solve. In such a situation we explore the findings systematically, to gather sufficient relevant evidence to formulate a specific diagnostic problem.

5.4 An Example

Let us consider a situation with two possible patient descriptions shown in figures 5.18 and 19.

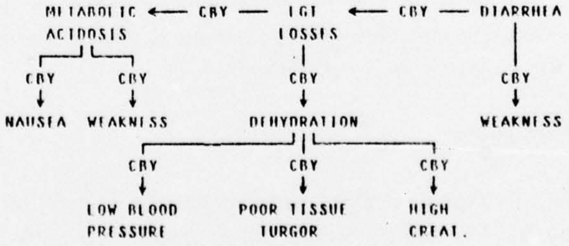


Fig. 5.18

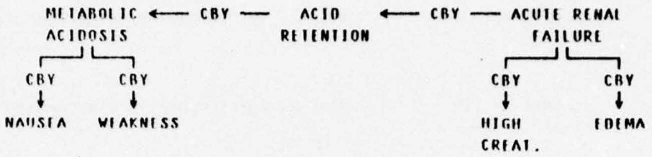


Fig. 5.19

The top level goal representation for the above patient is shown in figure 5.20.

```
GOAL 1
 PRIMARY-GOAL: (DIFFERENTIATE DIARRHEA ACUTE-RENAL-FAILURE)
 CONTEXT: (CAUSE-OF METABOLIC-ACIDOSIS)
 EXPECTATIONS:
    (OR
       LIKELY    [DIARRHEA
                      SEVERITY: SEVERE
                      DURATION: GREATER-THAN TWO DAYS
       POSSIBLE: [ACUTE-RENAL-FAILURE
                      SEVERITY: MODERATE
                      DURATION: GREATER-THAN ONE WEEK])
       SUBGOALS: GOAL 2
```

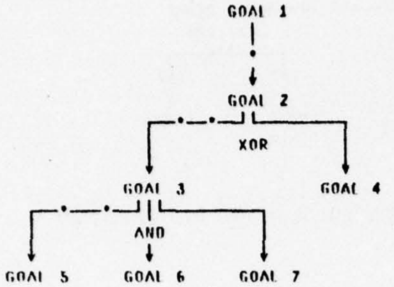
Fig. 5.20

We can differentiate between the diarrhea and acute-renal-failure by finding out the state of hydration of the patient.

```
GOAL 2
 PRIMARY-GOAL: (DIFFERENTIATE DEHYDRATION EDEMA)
 CONTEXT: (STATE-OF HYDRATION)
 EXPECTATION: (OR
                  LIKELY   [DEHYDRATION
                               SEVERITY: SEVERE]
                  POSSIBLE [EDEMA
                               NATURE: (OR GENERAL PEDAL)
                               SEVERITY: MODERATE])
SUBGOALS: (XOR GOAL 3 GOAL 4)
```

Fig. 5.21

This goal can be achieved by either confirming dehydration or edema. A graphic representation of complete goal structure for this situation is shown in figure 5.22



```
GOAL 1: (DIFFERENTIATE DIARRHEA ACUTE-RENAL-FAILURE)

GOAL 2: (DIFFERENTIATE DEHYDRATION EDEMA)

GOAL 3: (CONFIRM DEHYDRATION)
GOAL 4: (CONFIRM EDEMA)

GOAL 5: (CONFIRM LOW-BLOOD-PRESSURE)
GOAL 6: (CONFIRM POOR-TISSUE-TURGOR)
GOAL 7: (CONFIRM MODERATELY-ELEVATED-CREATININE)
```

Fig 5.22

5.5 Control Flow

In the section above we have noted that at the time some information is requested, the problem solver has a hierarchy of goals or a goal stack representing the downward (depth-first). locus of control flow (shown with "*" in fig. 5.6). We have also noted that each subgoal in the goal stack is associated with expectations. The expectations associated with a goal can be viewed as a predicate which must be satisfied when the goal is attained and fails if the goal is no longer attainable. More specifically the expectation predicate may evaluate to one of four values; (i) Success, (ii) Partial Success, (iii) Failure or (iv) Contradiction.

After every question the expectation of the immediate goal is evaluated. If the result is a success, then control is returned to the superior, with an indication of success. The evaluation results in partial success if the information gathered is not certain enough to satisfy the expectations but can be interpreted in a way that will lend support to the immediate goal. On the other hand if the evaluation results in a failure, the problem solver tries to look for an *excuse* or a caveat that can cause the particular goal to be negated without contradiction. This mechanism allows us to distinguish apparent contradiction from real contradiction, providing the information acquisition process some robustness. More importantly, it prevents erroneous evaluation of hypotheses by preventing apparently contradictory information from corrupting the patient description. For example, let us consider a patient suffering from a Urinary Tract Infection (UTI) who has recently received antibiotic therapy for some unrelated ailment. Now let us look at the program at a point where UTI is its leading hypothesis, it has requested information about the result of the urine culture and it is told that the urine culture is negative. Let us also assume that the program is not aware of the antibiotic therapy. If the program now evaluates the UTI-Hypothesis, the negative urine culture would almost always cause the hypothesis to be rejected. Now as UTI is no longer its leading hypothesis, the next question, about the recent use of antibiotics, which could explain the negative finding, will not be asked. To avoid such situations the program should delay the evaluation of the current hypothesis whenever an unexpected finding is encountered and should explore this finding in greater detail. If an excuse is found, the current goal may result in success or partial success. If no excuse is found, but the discrepancy is not very serious, it results in failure. In case of failure, the expectations of the immediate superior goal are evaluated and the above-described process is repeated from there. On the other hand, if a major discrepancy is found, there is some serious error in the patient description or in the problem formulation. This situation is handled with the help of a special contradiction handler, which identifies the scope of the contradiction by backtracking through the goal stack to find a point where the contradiction does not affect the formulation of superior goals. It also invokes a debugging program that modifies the patient description in the light of the new finding. If the resultant change in the patient model does not change the top level objective of the problem solver significantly, the program continues with the same problem structure; otherwise, a new problem is formulated.

The process of back-tracking is computationally expensive. Therefore, commonly occurring contradictions and heuristics for recovering from them are generally precompiled in the knowledge base of the program. Availability of this information allows clinicians (and the program) to recover from anticipated contradictory situations effectively without resorting to back-tracking explicitly.

The new information gathered during problem solving must continually be added to the patient description. This process is considerably simplified because the incoming information is consistent with the patient profile and its context is well defined. The use of the goal structure provides a strong focus to the question-asking behavior of the program and provides us with a mechanism to shift from "Global" to "Local" problem solving (a shift observed in the behavior of clinicians [36, 21, 19]). It also makes the reason for question asking and the context in which the question is being asked explicit, allowing us to explain if necessary why a particular question is asked and the effects of various answers to the question.

5.6 Summary

In this chapter we have discussed the process of problem identification and formulation, diagnostic strategies, goal structures representing the plan for problem solving, and the use of expectations in directing the flow of control.

The diagnostic problems are generated by identifying the places where two or more hypotheses differ from one another in the interpretation of the findings. We have noted in chapter 3 that a hypothesis has multiple interpretations if it has more than one superior hypothesis. All the problems identified above form the problem-set used in formulating a top level diagnostic goal for the problem solver. The problem solver generates a goal structure representing a tree structured plan for problem solving by decomposing this goal into subgoals recursively. The goal structure also identifies the parts of the patient description relevant for problem solving and the context in which the incoming information is evaluated.

The decomposition of the diagnostic goal into subgoals is done using various diagnostic strategies based on the protocol analyses by Miller [21], Kassirer [19] and Elstein [9]. These strategies and appropriate situations for invoking them are discussed in section 5.3. The strategies are Confirm, Rule-Out, Differentiate, Group-and-Differentiate, Refine and Explore. The choice of an appropriate strategy for a given situation is based on the number of hypotheses in the primary-goal and their belief values.

We note that clinicians have some prior expectations about the outcome of the problem solving effort. In the program, the expectations are described as a predicate that can evaluate to success, partial success, failure or contradiction. The expectation predicate of the immediate goal is evaluated after each step of problem solving and the result of this evaluation is used in directing the flow of control. The expectations are also used in determining the consistency of the incoming information. This allows us to identify apparent and real discrepancies before new information is assimilated into the patient description.

6. In Conclusion

For the intermediate term future, the significance of this research will be in terms of its contribution to the methodology of artificial intelligence based medical programs. It will ultimately enable us to build a program in the area of acid/base and electrolyte diagnosis and therapy which will include much of the knowledge of the best experts in that domain, but that stage of accomplishment probably lies beyond the end of the current phase of the project. At the moment, however, it is the development of appropriate methods and technical tools that is important if truly expert programs are eventually to be built.

We have described some of the deficiencies of the best programs which have been implemented with the current technology. The proposed AI techniques will allow us to address these problems by making available in the program explicit representations of all those components of the expert physician's decision making which we believe to be important to him. We have proposed a hierarchic representation of medical knowledge, permitting the description of causal and associational connections at several levels of detail, for expressing both general medical knowledge and specific information about the particular patient under consideration. We have introduced the notions of hypothesis and complete hypothesis and described how the physician's state of mind may be representable in the machine via these mechanisms. We have developed techniques for reformulating hypotheses based on the acquisition of additional information, and for selecting new information to seek based on the current hypothesis. We have also suggested a potentially useful method of deferring the need for detailed probabilistic computations while general diagnostic problem solving is in progress. Each of the above components represents one solution to important AI problems facing all researchers. Because the complexity of the application domain we have chosen is sufficiently high that we are forced to develop deep reasoning techniques to solve them, the results of our work should also be applicable to other efforts. For example, our notion of explicit hypotheses and the central role of their reformulation should be a significant addition to current diagnostic programs such as the Present Illness Program and Internist.

References

1. Betaque, N. E., and Gorry, G. A., Automating Judgmental Decision Making for a Serious Medical Problem, *Management Science* 17:8: B-421 - B-434 (April 1971).

2. Bleich, H. L., Computer-Based Consultation: Electrolyte and Acid-Base Disorders, *American Journal of Medicine* 53:285 (1972).

3. Bobrow, D. G., and Winograd, T., An Overview of KRL, a Knowledge Representation Language, Artificial Intelligence Laboratory, Stanford University, Technical Report AIM-293 (1976).

4. Cage, P. E., Carson, E. R., and Britton, K. E., Simulation of Renal Function in Man, *MEDINFO* 77 IFIP, North-Holland Publishing Company (1977).

5. Carlos, V., Pevny, E., and McMath, F., Computer Analysis of Blood Gases and Acid-Base States, *Computers and Biomedical Research* |4: 623-633 (1971).

6. Davis, R., Applications of Meta Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases, Ph.D. thesis, Stanford Artificial Intelligence Laboratory Memo 283 (1976).

7. Davis, R., Buchanan B., and Shortliffe, E., Production Rules as a Representation for a Knowledge-Based Consultation Program, *Artificial In'elligence* 8:15-45 (1977).

8. Duda, R. O., Hart, P. E., and Nilsson, N. J., Subjective Bayesean methods for rule-based inference systems, *Proceedings of the 1976 National Computer Conference,* AFIPS Press (June 1976).

9. Elstein, A. S., Shulman, L. A., and Sprafka S. A., *Medical Problem Solving,* Harvard University Press (1978).

10. Feigenbaum, E. A., and Feldman, J., *Computers and Thought* Edt. McGraw-Hill Inc. (1963).

11. Feinstein, A. R., Clinical Biostatistics XXXIX. The haze of Bayes, the aerial palaces of decision analysis, and the computerized Ouija board, *Clinical Pharmacology and Therapeutics* 21:4:482-496 (April 1977).

12. Forrester, J. W., *Industrial Dynamics,* M.I.T. Press, Cambridge (1961).

13. Gaines, B. R., Foundations of fuzzy reasoning, *Int. J. Man-Machine Studies* 8:623-668 (1976).

14. Goldberg, M., Green S., Moss, M. L., Marbach, C. B., and Garfinkel, D., Computer-based Instruction and Diagnosis of Acid-Base Disorders, *JAMA* 223:3:269-275 (1973).

15. Gorry, G. A., and Barnett, G. O., Sequential Diagnosis by Computer, *JAMA* 205:12:141-146 (1968).

16. Gorry, G. A., Kassirer, J. P., Essig, A., and Schwartz, W. B., Decision Analysis as the Basis for Computer-Aided Management of Acute Renal Failure, *The American Journal of Medicine* 55:473-484 (1973).

17. Gorry, G. A., Silverman, H., and Pauker, S. G., Capturing Clinical Expertise: A Computer Program that Considers Clinical Responses to Digitalis, *American Journal of Medicine* 64:452-460, (March 1978).

18. Gorry, G. A., Pauker, S. G., and Schwartz, W. B., The Diagnostic Significance of the Normal Finding, *N Engl J Med* 298:486 (1978).

19. Kassirer, J. P., and Gorry, G. A., Clinical Problem Solving: A Behavioral Analysis, *Ann. Int. Med.* 89:245, (1978).

20. Kuipers, B. J., Representing Knowledge of Large-Scale Space, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI-TR-418 (1978).

21. Miller, P. B., Strategy Selection in Medical Diagnosis, Project MAC, Massachusetts Institute of Technology, Technical Report TR-153 (September 1975).

22. Minsky, M., Semantic Information Processing, (Ed.) MIT Press, Cambridge, (1968).

23. Minsky, M., A Framework for Representing Knowledge *The Psychology of Computer Vision* Edt. Patric Winston, McGraw-Hill,(1975).

24. Newell, A., and Simon, H. A., *Human Problem Solving*, Prentice Hall (1972).

25. Newell, A., Artificial Intelligence and the Concept of Mind, in Schank, R. C., and Colby, K. M, eds., *Computer Models of Thought and Language*, W. H. Freeman and Co. (1973).

26. Pauker, S. G., and Kassirer, J. P., Therapeutic Decision Making: A Cost-Benefit Analysis, *N Engl J Med* 293:229-234 (July, 1975).

27. Pauker, S. G., Gorry, G. A., and Kassirer, J. P., and Schwartz, W. B., Toward the Simulation of Clinical Cognition: Taking a Present Illness by Computer, *The American Journal of Medicine* 60:981-995 (June 1976).

28. Pauker, S. G., and Szolovits, P., Analyzing and Simulating Taking the History of the Present Illness: Context Formation, in Schneider/Sagvall Hein, eds., *Computational Linguistics in Medicine*, North-Holland (1977).

29. Pauker, S. G., and Kassirer, J. P., Clinical Applications of Decision Analysis: A detailed illustration, *Seminars in Nuclear Medicine* (Oct. 1978).

30. Pople, H. E., Jr., Artificial-Intelligence Approaches to Computer-Based Medical Consultation, IEEE Intercon Conference (1975).

31. Pople, H. E., Jr., Myers, J. D., and Miller, R. A., DIALOG: A Model of Diagnostic Logic for Internal Medicine, *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence*, available from the Artificial Intelligence Laboratory, Massachusetts Institute of Technology (1975).

32. Pople, H. E., Jr., Presentation of the INTERNIST System, Proceedings of the A.I.M Workshop, Rutgers Research Resource on Computers in Biomedicine, Rutgers University (June 1976).

33. Pople, H. E., Jr., The Formation of Composite Hypotheses in Diagnostic Problem Solving: an Exercise in Synthetic Reasoning, *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, available from the Department of Computer Science, Carnegie-Mellon University, Pittsburgh, PA 15213. (1977).

34. Richards, B., and Goh, A. E. S., Computer Assistance in the Treatment of Patients with Acid-Base and Electrolyte Disturbances, *MEDINFO 77* IFIP, North-Holland Publishing Company (1977).

35. Rieger, C., The Commonsense Algorithm as a Basis for Computer Models of Human Memory, Inference, Belief and Contextual Language Comprehension, in R. Schank and B. L. Nash-Webber, eds., Proceedings of a Workshop on Theoretical Issues in Natural Language Processing, Massachusetts Institute of Technology (June 1975).

36. Rubin, A. D., Hypothesis Formation and Evaluation in Medical Diagnosis, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI-TR-316 (January 1975).

37. Sacerdoti, E. D., A Structure for Plans and Behavior, Stanford Research Institute Technical Note 109, (1975).

38. Safran C. et al., Diagnostic planning and cancer management, Massachusetts Institute of Technology, Technical Report LCS-TR-169, (1976).

39. Schwartz, W. B., Medicine and the Computer: The Promise and Problems of Change, *N Engl J Med* 283:1257-1264 (1970).

40. Schwartz, W. B., Gorry, G. A., Kassirer, J. P, and Essig, A., Decision Analysis and Clinical Judgment, *The American Journal of Medicine* 55:459-472 (October 1973).

41. Shafer, G., *A Mathematical Theory of Evidence*, Princeton University Press (1976).

42. Shortliffe, E. H., and Buchanan, B. G., A Model of inexact reasoning in medicine, *Mathematical Biosciences* 23:351-379 (1975).

43. Shortliffe, E. H., *Computer Based Medical Consultations: MYCIN*, Elsevier North Holland Inc. (1976).

44. Smith, B. C., A Proposal for a Computational Model of Anatomical and Physiological Reasoning, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, AI-Memo 493 (1978).

45. Swartout, W. R., A Digitalis Therapy Advisor with Explanations, Laboratory for Computer Science, Massachusetts Institute of Technology, Technical Report TR-176 (February 1977).

46. Szolovits, P., and Pauker, S. G., Research on a Medical Consultation System for Taking the Present Illness, *Proceedings of the Third Illinois Conference on Medical Information Systems*, University of Illinois at Chicago Circle (November 1976).

47. Szolovits, P., Personal Communication (1977)

48. Szolovits, P., Pauker, S. G., Categorical and Probabilistic Reasoning in Medical Diagnosis, *Artificial Intelligence* 11,:115-144 (1978).

49. Szolovits, P., Coordinating the Use of Categorical and Probabilistic Reasoning, *Proceeding of the 1978 Computer Laboratory Health Care Resources Workshop*, Texas Technical University School of Medicine, Lubbock, Texas (January 1978).

50. Szolovits, P., The Lure of Numbers--How to Live With and Without Them in Medical Diagnosis, *Proceeding of the Colloquium in Computer-Assisted Decision Making using Clinical and Paraclinical (Laboratory) Data*, 1978.

51. Tversky, A., and Kahneman, D., Judgment under Uncertainty: Heuristics and Biases, *Science* 185:1124-1131 (September 1974).

52. *Websters New Collegiate Dictionary*, A Merriam-Webster (1977)

53. Weiss, S. M., A System for Model-Based Computer-Aided Diagnosis and Therapy, Ph.D. Thesis, Computers in Biomedicine, Department of Computer Science, Rutgers University, CBM-TR-27-Thesis (June 1974).

54. Weiss, S. M., Presentation at the Third Illinois Conference on Medical Information Systems, University of Illinois at Chicago Circle (November 1976).

55. Weiss, S. M., Kulikowski, C. A., Amarel, S., and Safir, A., A Model-Based Method for Computer-Aided Medical Decision-Making, *Artificial Intelligence* 11: 145-172 (1978).

56. Zadeh, L. A., Fuzzy sets, *Information and Control* 8:338-353 (1965).

DATE

FILMED

8–8